DOCUMENT RESUME

ED 233 693 IR 010 783

AUTHOR Cohen, Vicki Blum
TITLE A Learner-Based Evaluation of Microcomputer Software.
PUB DATE Apr 83
NOTE 28p.; Paper presented at the Annual Meeting of the American Educational Research Association (Montreal, Canada, April 11-15, 1983).
PUB TYPE Reports - Research/Technical (143) -- Speeches/Conference Papers (150) -- Tests/Evaluation Instruments (160)

EDRS PRICE MF01 Plus Postage. PC Not Available from EDRS.
DESCRIPTORS Comparative Analysis; *Computer Assisted Instruction; *Computer Programs; *Educational Media; Elementary Secondary Education; *Evaluation Methods; Media Research; *Microcomputers; Questionnaires
IDENTIFIERS *Courseware Evaluation; *Learner Based Evaluation Systems

ABSTRACT

A project involving the evaluation of microcomputer courseware being marketed for school use was conducted in the Microcomputer Resource Center at Teachers College, Columbia University, during 1982 with funding from the Exxon Education Foundation. Every courseware package was evaluated by a team which included a subject matter expert, an instructional designer or educational psychologist, a practicing teacher from the surrounding urban and suburban areas, and the project director. Learner-based feedback was also incorporated by trying out the program on learners for whom the courseware was designed and using the information collected from this process to verify or refute the findings of the evaluation team. The evaluations were then compiled into a profile system for dissemination to school districts by the Educational Products Information Exchange (EPIE) Institute. This paper describes the project's theoretical framework and the evaluation procedure. Results of the evaluation of 29 courseware programs are presented, followed by an extensive list of conclusions drawn from these results. Present day microcomputer software is compared to the software evaluated in a similar study conducted in 1981. An analysis of where the field should be heading and recommendations for the future are also given. A sample form used to collect learner-based feedback concludes the paper. (Author/ESR)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

This document has been reproduced as received from the person or organization originating it.

Minor changes have been made to improve reproduction quality.

Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

ED233693

A Learner-based Evaluation of Microcomputer Software

by

Vicki Blum Cohen

Presented at the annual meeting of the AERA
Montreal, Canada; 1983

Printed in the U.S.A.

IR010783

"PERMISSION TO REPRODUCE THIS MATERIAL IN MICROFICHE ONLY HAS BEEN GRANTED BY

Vicki Blum Cohen

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

ABSTRACT

A Learner-based Evaluation of Microcomputer Software

by Vicki Blum Cohen

This paper describes a funded project by Exxon Education Foundation that took place in the Microcomputer Resource Center at Teachers College, Columbia University during the year of 1982. The project was involved in evaluating microcomputer courseware being marketed for school use. An integral part of the project was incorporating learner-based feedback into the evaluation process by trying the program out on those learners the courseware was designed for. This information was then used to verify or refute the findings of the evaluation team and was included in the final report. The evaluations were compiled into a "profile" system and disseminated by the Educational Products Information Exchange (EPIE) Institute to school districts across the country.

The results of this project are reported from the data collected during the evaluation. A comparison of the present day market of microcomputer software is compared to a similiar study conducted in 1981 by the same author. An analysis of where the field should be heading and recommendations for the future are given.

010783

Introduction

This paper describes a funded project by Exxon Education Foundation that took place in the Microcomputer Resource Center at Teachers College, Columbia University during the year of 1982. The project was involved in evaluating microcomputer courseware being marketed for school use. An integral part of the project was incorporating learner-based feedback into the evaluation process by trying the program out on those learners the courseware was designed for. This information was then used to verify or refute the findings of the evaluation team and was included in the final report. The evaluations were compiled into a "profile" system and disseminated by the Educational Products Information Exchange (EPIE) Institute to school districts across the country.

Theoretical Framework

With the proliferation of microcomputers into the classrooms of schools, there has been a commensurate proliferation of educational software which is starting to flood the market. Software and courseware have become the newest phenomenon in the publishing industry. While a year ago most publishing companies were watching the microcomputer market with trepidation and hesitation, today they are committed to producing software products and acquiring a part of a fertile and promising market.

The boom in software sales has also created a variety of new companies, most of which initially started as small cottage industries, producing software in their livingrooms and dens. The number of new companies that constantly emerge is in amazing proportions and daily some new product from an unknown firm emerges on the market.

The lack of quality software on the market has been documented (Blum Cohen, 1982). Much of the software produced is not instructionally effective and does not utilize the full potential of the technology. There is a great need for the evaluation of software which provides consumers and producers with qualitative information about the products. At this point in time it is necessary to positively direct the market toward a direction of quality and educational effectiveness.

Recognizing the need for such evaluation projects, there has been much difficulty in developing standards and evaluation procedures (Roblyer, 1981). There appears to be widely differing opinions about what the "ideal courseware product" should be, and what we should be aiming for in the way of courseware of the future. As criteria emerge, the market changes and new products with unique features are produced, making the criteria somewhat obsolete. It is therefore important that any guidelines and criteria be flexible, fluid and valid indicators which reflect the current state of a changing market.

Criteria and guidelines for evaluating software have been developed in the past year (Blum Cohen, 1982; Gagne, Wager and Rojas, 1981; Heck, Johnson and Kansky, 1981; MicroSoft, 1982). Although there may be disagreement about many of the issues, there is a fundamental similarity that runs among them concerning many of their criterion items. Most important is that standards are starting to exist which are defining this new technology in terms of educational effectiveness. It is mandatory that evaluation projects occur so that these standards can be utilized, tried, and altered according to the market. They must not stand in isolation, as rigid theoretical constructs, but be allowed to grow, change and be revised to reflect current standards.

Two extremely important sources of information that must be tapped during the evaluation process to ensure that the criteria are significant and that the evaluations are valid are 1) a practicing teacher and 2) a learner that the product was designed for. By using a classroom teacher as an evaluator the process becomes a closer approximation to the actual setting where the software product will be used. The teacher/evaluator can provide essential input about the validity of the program in a classroom environment and judge whether the product would actually work with students and teachers. S/he would provide invaluable comments about teacher training, integrating the program into the curriculum, ease of implementation, and pedagogical concerns. The learner provides invaluable insight into how the software product will be received and can validate many of the evaluators'

findings. Trying the program out with learners allows the evaluation team to subjectively judge the instructional effectiveness of the product by attaining information about motivational factors, likes and dislikes, and student preferences regarding learning. There is no other way of attaining such insight and it forces the evaluation team to utilize a much more applied, almost intuitional approach to evaluation.

The evaluation project described in this paper is based upon the above considerations and will describe them in further detail.

The Evaluation Procedure

The evaluation project at the Microcomputer Resource Center, Teachers College tried to address many of these issues by establishing a learner-based evaluation procedure. A similar study had been conducted in 1981 at the MRC (Blum Cohen, 1982) which had evaluated the first educational software products to emerge on the market for school use. This previous study was one of the first evaluation projects to be conducted and helped to establish guidelines and standards for this new field; however, two major criticisms had been noted: 1) it lacked a practicing teacher as part of the evaluation team, and 2) learner-based information had not been collected to validate the findings.

In response to these criticisms both factors became significant considerations when developing the evaluation procedure reported in this paper. During recruitment of evaluators, care was taken to include practicing teachers from

the surrounding urban and suburban areas. This became a mutually beneficial practice and provided excellent training to those teachers who were involved. They gained invaluable knowledge and expertise which they could bring back to their respective schools, while the project gained a practical and applied orientation to evaluation. Oftentimes it was the teacher who provided the necessary insight into a program by merely saying, "This is the type of program my kids would love and I could easily use it in my classroom." At that point, no matter how much dissent or disagreement existed about a particular program, the evaluators had to drop their theoretical constructs and intellectual pride and just listen. If a "boring, traditional drill and practice program" was found to be of value to a 5th grade New York City Public School teacher, who could argue and say the program was worthless? This type of input was so necessary as it brought many issues into perspective.

Training of evaluators consisted of handing them the form, giving them a brief explanation of the important concepts and terminology such as "feedback," "branching," "user control," and then having them evaluate a short simple software program using the form. As Project Director I would then read it over, mark down any comments I had, and meet with each evaluator individually. At this point the evaluator normally had many questions and needed clarification on various issues. When I felt each evaluator was competent and knowledgeable about the aspects of microcomputer software, he or she became part on an evaluation team and was assigned a program to evaluate.

During orientation each evaluator had been instructed to write down his or her background, area of expertise, and type of programs each would like to evaluate. This information became the basis for deciding who was to be included in each evaluation team. Every courseware package was evaluated by a team which included a subject matter expert, an instructional designer (or educational psychologist) who was knowledgeable about microcomputer software, a practicing teacher and myself, the Project Director. There were usually many such teams functioning at once and when a courseware package was completed, an evaluator would be assigned to a new team which was to evaluate another program.

This type of teamwork was found to be very beneficial and contributed toward a healthy, well-rounded evaluation. It is highly recommended that a small team consisting of evaluators from these four different backgrounds be used in evaluation projects. The three core evaluators - the subject matter expert, the instructional designer and the teacher - would individually evaluate each program in great depth and write up a detailed report. As the Project Director, I would look at each program long enough to become familiar with it and I would jot down detailed notes about its benefits and its flaws. I maintained the consistency throughout each team and contributed a knowledgeable background and guiding force throughout the project.

After all four team members had evaluated the courseware package a debriefing session was held. This session became the arena where all four personalities could meet to discuss, confront, and argue. It was my job as Project Director to try to compile a final report based on the four differing opinions. If no consensus could be reached this was noted in the report. Again, the teacher was found to be invaluable by offering insights and opinions about issues the other three could only guess at.

The last phase in the evaluation process was gathering learner-based information. Once the debriefing sessions were completed, the evaluators were asked if they knew any students who were the age or level that the product was designed for. If so, they were requested to bring them in. With this pool of students (which was fairly large due to the teachers) plus those students who attended Teacher's College in special programs, each courseware program was tried out with a learner and data was systematically collected using a form (see Appendix). This phase often turned out to be the most fun, the most interesting and the most informative. We always tried to validate our assumptions by purposely choosing a particular type of student or specific part of a program to try out. Therefore, the evaluators would usually have definite question in mind before the actual tryout. For example, one program we evaluated specified its target audience to include four year olds. During the debriefing sessions, evaluators felt strongly that the program was not appropriate for the average four year old. We therefore tried the program out

with three different four year olds and our assumptions were verified. In addition, we could also recommend how the program might be adapted to include this age range. In a similar fashion, evaluators found one program to be very confusing in how it taught multiplication skills. We brought in two sixteen year olds who shortly confirmed our assumptions.

During the learner try-out evaluators tried not to interfere with the students' session on the microcomputer, their role being that of an unobtrusive observer collecting data. After each session, which lasted from 1/2 hour to an hour, the evaluator would question the student using the form as a guideline. If necessary, specific questions about particular aspects of the program were also asked, i.e. were the graphics distracting, could you read the screen all the time, did you like to read stories on a screen? These responses were recorded and later used in the reports.

As a final step, the Project Director was required to take all of this information - the three evaluators' forms and the learner-feedback forms - and write up a composite report about the product.

This report was then presented to the E.P.I.E. editor who in turn transformed it into a final copy. Evaluators always received this final copy for revisions and comments before it went to the publisher.

This evaluation procedure was found to be very helpful in establishing a well-rounded report. It is a procedure that school districts could easily implement and it would help to increase their assurance that they are buying quality courseware programs specifically suited to their individual needs.

The following results can be noted regarding the instructional software programs evaluated in this project. These results are based upon the final report that was compiled using the evaluators' forms and the learner-based information.

Results

Table 1 represents a breakdown of the results by each criterion item found on the form used in this project. From this table it is possible to determine how many programs attended to specific design considerations. It must be noted that the total number of courseware programs included for analysis in this report is 29 and all percentages are based upon this N. This percentage does not reflect the total market but only what was evaluated in this project in the year of 1982.

Table 1

Breakdown of Results by Item

Criteria Item	%	N
1. Subject Area		
Spelling	7	2
Language	7	2
Social Studies	7	2
Science	3.5	1
Reading	17	5
Math	41	12
Logic/Problem Solving	17	5
2. Curriculum Role		
Supplemental	89.6	26
Basic Course	10.6	3
3. Mode of Interaction*		
Drill and Practice	62	17
Tutorial	24	7
Game	45	13
Simulation	10	3
4. Grouping*		
Individual	89	20
Pairs	20.6	6
Small Groups	13	4
Network	7	2
INTENTS		
5. Intended Users		
Not Specified	34	10
Specified	66	19
6. Target Audience*		
K-8	72	21
J.H.	24	7
H.S.	13	4
7. Objectives		
Inferred	66	19
Stated Clearly	34	10
8. Major Emphasis of Objectives		
Recall	62	18
Problem Solving	24	7
Concept Development	14	4
9. Field Testing		
Evidence reported clearly	3	1
None stated	97	28

*These categories were not all-inclusive and an evaluator could check as many as applied. Therefore % will not equal 100% and N will not equal 29.

Criteria Item	High %	High N	Middle %	Middle N	Low %	Low N
CONTENTS						
10. Appropriate match of content to student ability	31	9	66	19	3	1
11. Program scope reasonable	45	13	48	14	6	2
12. Appropriate readability	28	8	55	16	10	3
13. Graphics used appropriately	38	11	24	7	17	5
14. Program free of errors	66	19	34	10	-	-
15. Directions clear	34	10	55	16	10	3
16. Display is clear (frame formatting)	59	17	41	12	-	-
17. Use of examples, demonstrations	31	9	28	8	41	11
METHODS						
18. Approach appropriate for intended users	41	12	52	15	6	2
19. Approach enhances presentation	48	14	38	11	14	4
20. Teacher's Guide: suggestions valuable	6	2	34	10	60	17
21. Teacher's Guide: integrate program in the basic curriculum	14	4	6	2	80	23
22. Teacher's Guide: technical explanations clear and complete	48	14	31	9	20	6
23. User Control:						
Use of content menus	83	24	-	-	17	5
Review instructions	34	10	-	-	66	19
Exit at any time	66	19	-	-	34	10
Alter rate of presentation	14	4	-	-	86	25
HELP options available	3	1	-	-	97	28
24. Feedback:						
Is non-threatening	86	25	7	2	7	2
Remediates	3	1	31	9	66	19
Is immediate	83	24	3	1	14	4
25. Audio enhances the program	17	5	7	2	7	2
26. Program is easy to use	48	14	48	14	3	1
EVALUATION						
27. Overall placement is provided	14	4	-	-	86	25
28. Pretests provided	7	2	-	-	93	27
29. Unit/Mastery tests provided	24	7	-	-	76	22
30. Printed tests provided	6	2	-	-	94	27
31. Program branches for review	14	4	-	-	86	25
32. Program branches after a lesson	-	-	-	-	100	29
33. Program provides computerized management system	24	7	-	-	76	22
34. Management system is valuable	29	2	71	5	-	-
35. Management system is easy to use	43	3	57	4	-	-

The following conclusions can be drawn from the data analyzed in Table 1.

- o A majority of the programs available on the market are arithmetic programs;
- o Most programs are a drill and practice application and are almost exclusively (90%) used for supplementary use in the classroom;
- o Most programs (89%) are designed for individual use at the computer; 20% of the programs are designed to be used for pairs of students working together at the computer, 13% are designed for small groups and only 7% could be used in a classroom network;
- o The majority of the programs are targeted for elementary through junior high school (K-8) students;
- o The major emphasis of the objectives is recall;
- o Only one of the programs provide any hard data to substantiate field testing claims. 97% of the programs do not mention anything about field-testing. A few programs allude to developing the materials by trying them out with students and teachers but no evidence is reported;
- o Most of the programs' match of content to student ability is adequate;
- o 38% of the programs use graphics to their best advantage and were given high scores; 24% of the programs' graphics are adequate while 17% of the programs use graphics inappropriately.
- o 55% of the programs' directions for the student are not clear enough to warrent a high score; 10% of the programs' directions are rated low due to confusion and inadequate explanation; 34% of the programs have excellent directions;
- o Most of the programs do not use enough examples or demonstrations to help facilitate the instruction;
- o Most of the programs' approach to the content is appropriate for the intended users and this approach enhances the presentation;
- o Only 6% of the programs contain instructional suggestions in the accompanying Teacher's Guide that are seen to be valuable, while 60% of the programs are seen to be inadequate in this area.

- Only 14% of the Teacher's Guides contain any information on how to integrate the program into the basic curriculum while 80% of the programs do not even address this issue or do so in such a sketchy manner that it is seen as virtually useless;

- While not really addressing instructional or pedagogical concerns, most Teacher's Guides do contain adequate technical explanations on how to operate the microcomputer and run the software program;

- With regard to user control, most of the programs (83%) use some form of content menus to direct the user through the content; 34% of the programs allow the user to review instructions, while 66% of the programs allow the user to exit the program at any time. Only 14% of the programs allow the user to alter the rate of presentation and only 3% of the programs contain a HELP option for those students having difficulty.

- Feedback is non-threatening in 86% of the programs and is given immediately after a response in 83% of the programs. However only 3% of the programs provide any feedback which actually remediates the learner's response.

- Only 17% of the programs use audio which enhances the program; 52% of the programs do not utilize audio at all and 24% of the programs use it poorly, distracting and annoying the user or those who are nearby.

- Most of the programs are easy to use with only 3% being rated low due to extreme difficulty in utilizing it.

- With regards to evaluation, the following results can be reported:

  -Only 14% of the programs provide overall placement texts to facilitate placing students into the curriculum;

  -Only 7% provide pretests;

  -24% of the programs provide unit or mastery tests after a sequence of instruction;

  -Only 6% of the programs provide printed tests to supplement or replace computerized testing;

  -14% of the programs actually branch for review after a student enters a certain number of wrong responses. None of the programs branch after a lesson based upon results of student performance;

-Only 7 programs (24%) provide a computerized management system to track student progress and report the results to the teacher;

-Of these 7 programs which provide a management system, only 2 were found to be extremely valuable and only 3 were found to be very easy to use.

How Far Have We Come?

What do the above results mean when trying to assess the overall market of instructional software for the microcomputer? How can it all be put into perspective to make some sense? Perhaps the best way is to compare these results to a similar study that was done in 1981 (Blum Cohen) to highlight how the market has changed.

1) First of all, the most evident change in the market from 1981 to 1982 is the overwhelming amount of courseware packages to emerge. In 1981 a handful of publishers decided to try the market out and produced a few instructional software packages. Most of the initial impetus in the instructional software market came from cottage industry and lone programmers who wanted to make a quick dollar. This is no longer true. Most of the major educational publishing companies which have traditionally produced instructional materials are now involved (or about to be involved) in producing courseware. Furthermore, many of the cottage industries are no longer so small and inexperienced. Many have greatly increased their volume of sales and now produce a large variety of programs to sell. This has resulted in an enormous amount of software packages which have flooded the

market. There is no longer any shortage of courseware programs for educators to select from. Rather, the problem now lies in trying to determine what is quality software and how to evaluate and select software from the myriad of packages that exist.

2) The second most evident change is directly related to the first. Due to the greater amount of software that has emerged on the market, there is also a greater variety of subject areas that educators can choose from. In 1981 95% of the programs were arithmetic packages, while out of the 29 programs sampled in this study, 12 of them (45%) are arithmetic. 17% of the programs were reading which showed a significant increase over the previous year. Surprisingly the teaching of logic was highly represented and this was due predominately to one specific company which is producing high quality programs that teach logical problem-solving skills. Still fairly underrepresented was science, social studies, and language arts programs.

3) The curriculum role of the programs is still predominately supplemental and the majority of programs are still a drill and practice application. This has not changed since 1981 although there is a greater variety of tutorials, games and simulations on the market.

4) Most of the programs are still designed to be used in elementary to junior high school (K-8). However, there is a greater choice in the higher grades although many more programs need to be developed to accomodate the high school market.

5) Although the major emphasis of the objectives is still recall, more programs are attempting to teach problem-solving skills and concept development. Much of this effort is seen in one or two companies which are trying to design programs that challenge students and teach logic as opposed to seeing this improvement cut across the market to all producers.

6) In general, the graphic capability of software programs has greatly improved. There are many more programs out on the market which utilize colorful, exciting graphics that are embedded in the content. However, there are still many programs which do not use graphics at all and present the information in purely text format. Other programs use graphics poorly and could greatly improve their graphic capability.

7) The Teacher's Guide is still notoriously inadequate in most instances and no improvement is seen in their addressing pedagogic of classroom concerns.

8) There seems to be a bit more user control granted to the learner. Content menus and exiting the program at any time are the two most frequent devices used. Students are still not able to review instructions in most programs, alter the rate of presentation or call on a HELP function.

9) Feedback responses still do no remediate the wrong answers although most programs seem to be using feedback more appropriately.

10) Surprisingly, there has been little technological advancement in designing programs with a computerized management system. Only 7 out of the 29 programs provided a management system and they seemed no more elegant, sophisticated or advanced than the first program that appeared on the market. Few programs have cumulative records that are stored on a magnetic device for future retrieval and none of the programs branched a student to a lesson based upon performance.

Where Should We Be Heading?

From the above data the most glaring development in the market seems to be the enormous amount of instructional software packages that has innundated the field. With this greater amount of programs to choose from, there is an understandably greater selection of software programs in the different subject areas, for different target audiences, to suit different roles and to satisfy differing modes of interaction. However, the actual technological capabilities of these programs do not seem to be any more advanced nor does the actual instructional design seem to be any more improved than the first products that appeared in 1981. Instructional strategies are still not being incorporated into the design in a systematic fashion and certain capabilities seem to be rarely employed.

Admittedly with all of these new products emerging there are some producers who are exploring how this technology can be utilized and are developing high quality programs. In general, there seems to be a much greater awareness of what the microcomputer is and how it can be used for instructional purposes. The whole notion of feedback, user control, non-linear format of content and record-keeping devices are not such foreign terms anymore. However, whether these attributes are actually being utilized effectively in courseware is another matter.

The following recommendations will try to address many of the issues previously discussed here. They are made in hopes that producers of instructional software will attend to them so that the market will reflect no only a higher quality product but one that is advancing toward its full potential.

1) There need to be more programs in such subject areas as science, social studies, and language ares. These programs should be designed for students in middle school and especially high school where there is still a great lack of courseware packages.

2) Different modes of interacting with the computer other than drill and practice should be explored. Good tutorials which offer individualized instruction need to be developed. Simulations and gaming should be utilized more frequently.

3) Programs which are designed to be used by pairs or small groups of students working together need to be developed. The computer is particularly well-suited for social engagement and this type of instruction could greatly benefit classrooms.

4) There is still a great demand for those programs which teach critical thinking skills, problem-solving techniques and concept development. A few programs have emerged which are designed to do this. Such an attempt should cut across the market so that all producers are emphasizing higher-order skills in their insturctional design and not just mere recall of previously learned facts.

5) More sophisticated use of graphics is still needed so that the concepts taught are visually reinforced. Graphics should become an integral part of the content and be used as a motivational device.

6) There needs to be a much more effective use of examples and demonstrations. This instructional strategy is definitely lacking in most courseware. When used, examples and demonstrations help to teach concepts and reinforce appropriate responses. Unfortunately, it is rarely used in courseware programs.

7) The Teacher's Guide needs to be improved and carefully thought out. It should contain:

-Specification of the rationale, goals and especially behavioral objectives;

-A clear and concise statement of who the target audience is or who the program is designed for;

-What entry competencies are required for successful use of the program;

-Clear, concise and well-organized technical explanations on how to implement the program. This includes very detailed invormation on how to access and use the management system, if it is part of the program;

-Instructional suggestions and strategies for using the program in a classroom;

-Specific instructional activities to integrate the program into the curriculum;

-A scope and sequence chart would be helpful with larger curriculum courseware packages;

-Field-testing results or any documentation on learner verification studies in schools;

-Placement provisions on how to place students into the program;

-Supplemental activities and worksheets which can be used by students to help augment actual time spent on the computer;

8) Much more user control should be integrated into the instructional design of courseware programs. Students should have the chance to review instructions if needed, to exit the program at any time, to alter the rate of presentation and to call on HELP options.

9) Feedback should remediate students' wrong responses. By this it is meant that an alternate method of presenting the material should be employed after a student answers the problem incorrectly. Presenting the same instructional sequence again is not remediation, just review.

10) Why not utilize the audio component so it enhances the program? Audio could reinforce correct or incorrect responses. Care should be taken that there is a built-in option to turn this feature off when used in a classroom setting.

11) Placement tests and/or placement provisions should be provided in all programs. This would greatly facilitate ease of use for the teacher.

12) Testing should be incorporated into the design of a courseware program after each unit of instruction. Criterion scores should be based upon a fair representation of sample items, i.e. failure should not be based upon getting 3 out of 4 wrong but rather 7 out 10 wrong.

13) Actual branching is rarely employed in courseware programs. True branching occurs when the program offers different instructional sequences that a student is automatically placed into based upon level of performance. A student can then pursue his/her own level of instruction without ever being aware any branching is taking place.

Providing a "loop back" to a previous sequence for review is not true branching; nor is it branching when the program automatically places a student into the correct lesson each time she/he signs on. This capability is one of the greatest benefits of the computer and needs to be utilized in many more programs.

14) More programs should utilize computerized management systems in their design. These management systems should be easy to use, contain informative cumulative records on student progress for teachers to access, and pinpoint student deficiencies.

Summary

The need to evaluate microcomputer software is paramount given the proliferation of microcomputers in today's schools and the state of flux and change in the software market. Schools, consumers and producers need this type of information to help give direction to this new field and put it into perspective. The dissemination of the evaluations is crucial as well as reports about the project so that other agencies may pursue similar projects.

An important aspect of the evaluation is that the reviews reflect valid information. The project reported in this paper is unique in that it is one of the first evaluations to incorporate a practicing teacher as part of the team of evaluators, and that gathered learner-based information on each program to validate findings. This project was an attempt to take a step forward in evaluation by making it more responsive to learner's and consumer's needs.

It is hoped that school districts will use much of this information to develop their own evaluation projects. It is also hoped that this paper will help producers of instructional software to become more aware of the market and the direction it should be heading toward, thereby creating a higher quality product.

Student Name: _____ Courseware Program: _____

Grade: _____ Section/Level Used: _____

Age: _____ Date: _____

STUDENT OBSERVATION OF MICROCOMPUTER COURSEWARE USE

I. Physical Environment Information

Describe physical layout of the classroom including: where the computers are located; where the students are seated; and where the teacher(s) is (are) located. Use the following symbols:

/C/= Computer /_/= Desks X= Student T=Teacher [ ]= Blackboard

Comments: _____

_____

_____

_____

_____

_____

II. Ease of Use of Courseware Program

Student Checklist	Yes	Somewhat	No
1. Student has a problem manipulating the keyboard.	___	_____	___
Comments: _____			
_____			
2. Program holds student attention.	___	_____	___
Comments: _____			
_____			
3. Student has a problem following program's directions.	___	_____	___
Comments: _____			
_____			

	Yes	Somewhat	No
4. Student is self-sufficient and is not asking teacher many questions after initial introduction.	___	_____	___
5. Student has a problem reading the screen.	___	_____	___

Comments: _____

_____

IV. Student Questionnaire

1. Did you like this program? _____ Why or why not? _____

_____

2. Did you learn anything from it? _____ What did you learn (or why didn't you learn)? _____

_____

3. Could you follow the instructions all the time? _____

_____

4. Would you want to use this program again? _____ Why or why not? _____

_____

5. Did you like using the microcomputer? _____ Why or why not? _____

_____

6. What did you think of the graphics/pictures (if there were any)? _____

_____

7. Did you find the screen easy to read? _____

_____

III. Teacher Questionnaire

1. What type of student is this? _____

2. Does he/she do well in _____ subject area? _____

3. What are his/her strengths? _____

Weaknesses? _____